

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

LISTING OF CLAIMS:

1. (currently amended) A distributed data processing system comprising a plurality of servers and a plurality of clients connected to the servers for performing a distributed data processing process on an object to be controlled,

each of said servers comprising:

a database memory for storing a database which is updated by the distributed data processing process performed by said clients which alters data;

a replication trigger generator for generating a replication trigger based on ~~the updating of a change of data in~~ said database caused by the distributed data processing process performed by said clients connected to one of the servers;

an updating information transfer unit for transferring data updating information of said database to another one of the servers based on said replication trigger; and

a database updating processor for directly updating data in said database based on the data updating information transferred from the other server; and

~~an archive data memory for storing updating information of said database as archive data; wherein at least part of said database is recovered using said archive data.~~

2. (original) A distributed data processing system according to claim 1, wherein each of said clients comprises:

a connection information manager for managing connection information of a connection destination server to which the clients are connected; and

a connection information changer for changing the connection information of the connection destination server;

the arrangement being such that if any of said servers suffers a fault, said connection information is changed by said connection information changer, and the distributed data processing process performed by the clients connected to the server which suffers the fault is continued under the management of another normal one of the servers to which said connection information is changed.

3. (currently amended) A distributed data processing system according to claim 1, further comprising:

a backup processor for performing backup process at predetermined time intervals while said database is in operation;

a backup data memory for storing backup data produced by the backup process performed while said database is in operation; and

an archive data memory for storing updating information of said database as archive data;

wherein at least part of said database is recovered using said archive data;

wherein said database is recovered using said backup data and said archive data.

4. (currently amended) A distributed data processing system according to claim 2,
further comprising:

a backup processor for performing backup process at predetermined time intervals while
said database is in operation;

a backup data memory for storing backup data produced by the backup process
performed while said database is in operation; and

an archive data memory for storing updating information of said database as archive data;
wherein at least part of said database is recovered using said archive data;

wherein said database is recovered using said backup data and said archive data.

5. (original) A distributed data processing system according to claim 1, wherein said
servers comprise:

a server for managing one of the clients which is of a production management system
which is of the object to be controlled; and

a server for managing one of the clients which is of a process control system which is of
the object to be controlled.

6. (original) A distributed data processing system according to claim 5, wherein each of
said servers has independent settings of distributed data processing so that said database can be
independently processed in inserting, updating, or deleting data.

7. (currently amended) A method of processing data in a distributed data processing system having a plurality of servers and a plurality of clients connected to the servers for performing a distributed data processing process on an object to be controlled, comprising the steps of:

updating a database according to the distributed data processing process performed by said clients which alters data;

generating a replication trigger based on ~~the updating of~~ change of data in said database caused by the distributed data processing process performed by said clients connected to one of the servers;

transferring data updating information of said database to another one of the servers based on said replication trigger;

directly updating said database based on the updating information transferred from the other server; ~~and~~

~~storing the updating information of said database as archive data;~~

~~wherein at least part of said database is recovered using said archive data.~~

8. (currently amended) A method according to claim 7, further comprising the steps of:
determining a status of any of said servers, and if any of said servers suffers a fault,
changing a connection destination of the clients connected to the server which suffers the fault to another normal one of the servers; and

continuing the distributed data processing process performed by the clients connected to the server under the management of the other normal server.

9. (original) A method according to claim 8, further comprising the step of activating again said server suffering the fault to resume normal operation after completion of a restoring process, said step of activating again said server comprising the steps of:

- shutting off all the clients connected to said server;
- setting again information of the connection destination of the clients;
- connecting the clients to said server according to the set information; and
- resuming the distributed data processing process in a normal connection state.

10. (original) A method according to claim 7, further comprising the steps of:
performing a backup process at predetermined time intervals while said database is in operation and saving backup data produced by the backup process performed;

generating and saving archive data based on the updating information of the database which is generated after the backup process performed while said database is in operation has started; and

if one of said servers suffers a fault, copying said backup data of another normal one of the servers, and recovering the database from said archive data of the other normal server.

11. (original) A method according to claim 10, further comprising the step of:
copying said backup data while the clients are being continuously operated by said other
normal server.

12. (original) A method according to claim 8, further comprising the steps of:
performing a backup process at predetermined time intervals while said database is in
operation and saving backup data produced by the backup process performed;
generating and saving archive data based on the updating information of the database
which is generated after the backup process performed while said database is in operation has
started; and
if one of said servers suffers a fault, copying said backup data of another normal one of
the servers, and recovering the database from said archive data of the other normal server.

Claims 13.-15. (cancelled).

16. (previously presented) The distributed data processing system according to claim 1,
wherein the clients connected to any one of said servers are different from the clients connected
to another one of said servers.

17. (currently amended) The distributed data processing system according to claim 1,
wherein said database updating processor in each of said servers updates said database based on

the data updating information, the data updating information is generated upon an updating request from one of said clients connected to said server, and said database updating processor transfers the data updating information of said database to another one of said servers.

18. (currently amended) The distributed data processing system according to claim 17, wherein said database updating processor determines whether the data updating information is generated by said server or is transferred from the other one of said servers, and wherein when it is determined that the data updating information is transferred from the other one of said servers, a replication trigger generation inhibition is issued to said replication trigger generator.

19. (previously presented) A distributed data processing system according to claim 1, wherein the updating of the database occurs prior to the generating of the replication trigger.

20. (new) The system of claim 1, wherein the database updating processor polls for data updating information sent by another of the servers and wherein when the poll for updating information is successful, said database updating processor determines a presence of data changes in said updating information, and updates said database according to the data changes, and in response to the update, said replication trigger generator generates said replication trigger signal for transfer to a second one of the servers.

21. (new) The system of claim 20, wherein when the database updating processor does not determine the presence of data changes in said updating information, said replication trigger generator is inhibited from generating said replication trigger signal.

22. (new) The method of claim 7, wherein updating the database according to the distributed data processing comprises:

polling for data updating information sent by another of the servers, and wherein when the polling for updating information is successful, further determining a presence of data changes in said updating information, and updating said database according to the data changes, and wherein replication trigger generator is generated for transfer to a second one of the servers.

23. (new) The method of claim 22, wherein when said determining of presence of data changes indicates no presence of data changes, generating an inhibit signal to stop transfer to the second one of the servers.